CLAIMS

1. A method of downloading at least part of an application to an MPEG receiver/decoder, comprising the steps of:

dividing the application into a plurality of modules;

formatting each of the modules as a respective MPEG table, the tables having the same table identification ("TID") and respective different table identification extensions ("TID–extensions") other than a predetermined TID–extension;

generating a directory MPEG table for the modules having the same said TID and said predetermined TID–extension, the directory containing for each of the modules a name of that module and the respective TID–extension;

cyclically transmitting the directory MPEG table and the module MPEG tables in an MPEG bitstream; and

at the MPEG receiver/decoder:–

receiving the MPEG bitstream;

downloading that one of the MPEG tables having the predetermined TID–extension so as to download the directory MPEG table;

determining from the content of the directory MPEG table the TID–extensions of the module MPEG tables; and

downloading at least one of the module MPEG tables having the same TID as that of the downloaded directory MPEG table and a TID–extension determined from the downloaded directory MPEG table.

2. A method as claimed in claim 1, further comprising the steps of:

including in the transmitted directory MPEG table a version identification therefor; and

at the receiver/decoder:–

determining whether the version identification of the currently transmitted directory MPEG table is more recent than the version identification of the currently downloaded directory MPEG table, and if so repeating the steps of downloading the directory MPEG table, determining the TID–extensions, and downloading at least one of the module MPEG tables.

3. A method as claimed in claim 1 ~~or 2~~, wherein at least one of the module MPEG tables is formatted as a plurality of MPEG sections which are transmitted separately in the MPEG bitstream, each of the MPEG sections containing in a predetermined portion thereof an identification of that MPEG section in the MPEG table and an indication of the number of the sections in a MPEG table.

4. An MPEG receiver/decoder for use in performing part of the method of ~~any preceding~~ claim, comprising:

a receiver for receiving the MPEG bitstream;

a storage means; and

processing means which is programmed to cause that one of the received MPEG tables having the predetermined TID–extension to be downloaded to the storage means, to determine from the content of the directory MPEG table the TID–extensions of the module and MPEG tables, and to cause at least one of the module MPEG tables having the same TID as that of the downloaded directory MPEG table and a TID–extension determined from the downloaded directory MPEG table to be downloaded to the storage means.

5. A receiver/decoder as claimed in claim 4 ~~for use with the method of claim 2~~, wherein the processing means is programmed to determine whether a version identification of the currently received directory MPEG table is more recent than a version identification of the downloaded directory MPEG table, and if so to repeat the downloading of the directory MPEG table, determination of the TID–extensions and downloading of at least one of the module MPEG tables.

6. A receiver/decoder as claimed in claim 4 ~~or 5, for use with the method of claim 3~~, wherein the processing means is programmed to cause the MPEG sections to be repeatedly downloaded to the storage means until the processing means determines from the section identifications and the section number indication of the downloaded sections that all of the sections have been downloaded.

7. A receiver/decoder as claimed in ~~any of claims 4 to 6~~ claim 4, further comprising a

parallel port and/or a serial port arranged to receive an application formatted as at least one MPEG table.

8. An MPEG transmission system comprising:

means for dividing into a plurality of modules an application to be downloaded to an MPEG receiver/decoder;

means for formatting each of the modules as a respective MPEG table, the tables having the same TID and respective different TID-extensions other than a predetermined TID-extension;

means for generating a directory MPEG table for the models having the same said TID and said predetermined TID-extension, the directory containing for each of the modules a name of that module and the respective TID-extension; and

means for cyclically transmitting the directory MPEG table and the module MPEG tables in an MPEG bitstream.

9. A system as claimed in claim 8, further comprising means for generating a version identification for the directory MPEG table; and wherein the directory MPEG table generating means is operable to include in the directory MPEG table the generated version identification therefor.

10. A system as claimed in claim 8 ~~or 9~~, wherein the module formatting means is operable to format at least one of the module MPEG tables as a plurality of MPEG sections each containing in a predetermined portion thereof an identification of that MPEG section in that MPEG table and an indication of the number of the MPEG sections in that MPEG table.

11. A method of downloading data to an MPEG receiver/decoder, comprising the steps of:

generating a signature for the data to be downloaded;

encrypting the signature using a private key;

formatting the data to be downloaded, the encrypted signature and an identification for the private key as an MPEG table;

transmitting the MPEG table; and

at the receiver/decoder:–

receiving the MPEG table;

selecting one of a plurality of public keys in accordance with the key identification in the received MPEG table;

decrypting the encrypted signature in the received MPEG table using the selected public key to provide a decrypted signature;

generating a signature for the data in the received MPEG table; and

comparing the decrypted signature and the signature generated at the receiver/decoder for the received data.

12. A method as claimed in claim 11, further comprising the steps of downloading to the receiver/decoder an application having a signature encrypted using a private key having a predetermined key identification; running the application at the receiver/decoder to cause the receiver/decoder to receive a further key; storing the further key in an area of volatile memory of the receiver/decoder.

13. A method as claimed in claim 12, wherein during the step of running the application, the further key is supplied locally to the receiver/decoder.

14. A method as claimed in claim 13, wherein the further key is supplied to the receiver/decoder via a parallel port, serial port or smart card reader of the receiver/decoder.

15. A method as claimed in ~~any of claims 11 to 14~~ claim 12, further including the steps, at the receiver/decoder, of looking up, in a protected area of memory of the receiver/decoder, a validation flag for the selected public key, and inhibiting or aborting downloading of the data if the looked–up flag is not set.

16. A method as claimed in claim 15 ~~when dependent on any of claims 12 to 14~~, wherein, in the protected area of memory of the receiver/decoder, the private key having the predetermined key identification has a validation flag which can be changed by said

application, and an ability to receive such a further key is determined in dependence upon the state of that validation flag.

17. A method of downloading data to an MPEG receiver/decoder, comprising the steps of:–

generating a signature for the data to be downloaded;

encrypting the signature using a private key;

formatting the data to be downloaded, the encrypted signature and an identification for the private key as an MPEG table;

transmitting the MPEG table; and

at the receiver/decoder:–

receiving the MPEG table;

looking up, in a protected area of memory of the receiver/decoder, a validation flag for a public key corresponding to the private key identified in the received MPEG table; and

if the looked–up flag is set:–

decrypting the encrypted signature in the received MPEG table using the public key corresponding to be private key identified in the received MPEG table to provide a decrypted signature;

generating a signature for the data in the received MPEG table; and

comparing the decrypted signature and the signature generated at the receiver/decoder for the received data.

18. A method as claimed in ~~any of claims 11 to 17~~ claim 11, further including the steps of:–

generating a validation code for the data to be downloaded, the validation code being encrypted with the signature in the encryption step and being decrypted with the signature in the decryption step;

looking up a stored validation code in a protected area of memory of the receiver/decoder; and

comparing the looked–up validation code and the decrypted validation code.

19. A method of downloading data to an MPEG receiver/decoder, comprising the

steps of:-

generating a validation code for the data to be downloaded;

generating a signature for the data to be downloaded, or a part thereof;

encrypting the validation code and the signature using a private key;

formatting the data to be downloaded and the encrypted validation code and signature as at least one MPEG table;

transmitting the or each MPEG table; and

at the receiver/decoder:-

receiving the or each MPEG table;

decrypting the encrypted validation code and signature in the received MPEG tables(s) using a public key corresponding to the private key;

looking up a stored validation code in a protected area of memory of the receiver/decoder;

comparing the looked-up validation code and the decrypted validation code;

generating a signature for the data in the received MPEG table(s) or said part thereof; and

comparing the decrypted signature with the signature generated at the receiver/decoder for the received data.

20. A method as claimed in claim 18 ~~or 19~~, further including the step of inhibiting or aborting downloading of the data if, in the validation code comparing step, the looked-up validation code and the decrypted validation code do not match each other.

21. A method as claimed in ~~any of claims 11 to 20~~ claim 11, wherein the signature of the data to be downloaded is encrypted in a block of data including other data, with a selected offset between the start of the data block and the start of the signature, and the encrypted data block is decrypted in the decryption step at the receiver/decoder, and further including the steps, at the receiver/decoder, of looking up at least one stored offset in a protected area of memory of the receiver/decoder, and extracting the signature from the decrypted data block using said one looked-up offset from the start of the decrypted data block.

22. A method of downloading data to an MPEG receiver/decoder, comprising the steps of:–

generating a signature for the data to be downloaded;

including the signature and other data in a block of data with a selected offset between the start of the data block and the start of the signature;

encrypting the data block using a private key;

formatting the data to be downloaded and the encrypted data block as an MPEG table;

transmitting the MPEG table; and

at the receiver/decoder:–

receiving the MPEG table;

decrypting the encrypted data block in the received MPEG table using a public key corresponding to the private key;

looking up at least one stored offset in a protected area of memory of the receiver/decoder;

extracting the signature from the decrypted data block using said one looked–up offset from the start of the decrypted data block;

generating a signature for the data in the received MPEG table; and

comparing the signature extracted from the decrypted data block with the signature generated at the receiver/decoder for the received data.

23. A method as claimed in claim 21 ~~or 22~~, wherein said protected area of memory has at least two such stored offsets, and, if in the comparing step the extracted signature and the generated signature do not match, further including the steps of repeating the looking–up, extracting and comparing steps using another of the stored offsets.

24. A method as claimed in claim 21 ~~any of claims 21 to 23~~, wherein at least some of said other data in the block of data is dummy or arbitrary data.

25. A method as claimed in claim 11 ~~any of claims 11 to 24~~, wherein the data is downloaded as a plurality of modules of the data, and including the steps of:–

generating a module signature for each module of data to be downloaded;

formatting the modules of data as respective module MPEG tables;

generating a directory including an identification of each module MPEG table and the respective signature, the directory being the subject of the signature generating step ~~of any of claims 11 to 24~~; and

at the receiver/decoder:–

generating a respective module signature for each of the modules in the received module MPEG tables; and

comparing each module signature in the received directory MPEG table with the respective module signature generated at the receiver/decoder.

26. A method of downloading a plurality of modules of data to an MPEG receiver/decoder, comprising the steps of:–

generating a module signature for each module of data to be downloaded;

formatting the modules of data as respective module MPEG tables;

generating a directory including an identification of each module MPEG table and the respective signature;

generating a directory signature for the directory;

encrypting the directory signature using a private key;

formatting the directory and the encrypted directory signature as a directory MPEG table;

transmitting the directory and module MPEG tables; and

at the receiver/decoder:–

receiving the directory and module MPEG tables;

decrypting the encrypted directory signature in the received directory MPEG table using a public key corresponding to the private key;

generating a directory signature for the directory in the received directory MPEG table;

comparing the decrypted directory signature and the directory signature generated at the receiver/decoder;

generating a respective module signature for each of the modules in the received module MPEG tables; and

comparing each module signature in the received directory MPEG table with the respective module signature generated at the receiver/decoder.

27. A method as claimed in claim 25 ~~or 26~~, further including the step of inhibiting or aborting downloading of such a module of the data if, in the module signature comparing step, the module signature in the received directory MPEG table and the respective module signature generated at the receiver/decoder for that module do not match each other.

28. A method as claimed in ~~any of claims 11 to 27~~ claim 11, further including the step of inhibiting or aborting downloading of the data if, in the comparing step(s), the or each decrypted signature and the generated signature do not match each other.

29. An MPEG receiver/decoder for use in performing part of the method of claim 11, comprising:
means for receiving such MPEG tables;
means for storing a plurality of public keys and an identification for each of the public keys; and
processing means which is programmed to select one of the stored public keys in accordance with the key identification in the received MPEG table; to decrypt the encrypted signature in the received MPEG table using the selected public key to provide a decrypted signature; to generate a signature for the data in the received MPEG table; and to compare the decrypted signature and the signature generated at the receiver/decoder for the received data.

30. A receiver/decoder as claimed in claim 29, wherein the key storing means is provided by ROM.

31. A receiver/decoder as claimed in claim 29 ~~or 30~~, wherein the identification for each of the public keys is provided by the storage location of that public key in the key storing means.

32. A receiver/decoder as claimed in ~~any of claims 29 to 31 for use in the method of claim 14~~ claim 29, further including an area of volatile memory, and wherein the processing means is operable to download an application having a signature encrypted using a

private key having a predetermined key identification, to run the application to cause the receiver/decoder to receive a further key, and to cause the further key to be stored in the area of volatile memory.

33. A receiver/decoder as claimed in claim 32, further including means to receive such a further key which is supplied locally to the receiver/decoder.

34. A receiver/decoder as claimed in claim 33, wherein the further key receiving means is provided by a parallel port, serial port and/or smart card reader of the receiver/decoder.

35. A receiver/decoder as claimed in ~~any of claims 32 to 34~~ claim 32, wherein the volatile memory is provided by RAM.

36. A receiver/decoder as claimed in ~~any of claims 29 to 35 for use in the method of claim 15~~ claim 32, further including a protected area of memory for storing a validation flag for each of at least some of the public keys, and wherein the processing means is programmed to look–up, in the protected area of memory, the validation flag for such a selected public key, and to inhibit or abort downloading of the data if the looked–up flag is not set.

37. A receiver/decoder as claimed in claim 36 ~~when dependent on any of claims 32 to 35~~, further including a protected area of memory for storing a validation flag for the private key having the predetermined key identification, and wherein the processing means is operable when running said application to change that validation flag and is operable to enable the further key to be so stored in dependence upon the state of that flag.

38. An MPEG receiver/decoder for use in performing part of the method of claim 17, comprising:

means for receiving such MPEG tables;

means for storing a public key and an identification for the public key; and

a protected area of memory for storing a validation flag for the public key; and processing means which is programmed to look-up, in the protected area of memory of the receiver/decoder, a validation flag for the public key corresponding to the private key identified in the received MPEG table; and, if the looked-up flag is set, to decrypt the encrypted signature in the received MPEG table using the public key corresponding to be private key identified in the received MPEG table to provide a decrypted signature, to generate a signature for the data in the received MPEG table; and to compare the decrypted signature and the signature generated by the receiver/decoder for the received data.

39. A receiver/decoder as claimed in ~~any of claims 36 to 38~~ claim 36, wherein the memory for storing the key validation flag(s) is provided by rewritable non-volatile memory.

40. A receiver/decoder as claimed in ~~any of claims 36 to 39~~ claim 36, and in the case where a plurality of such public keys are stored, wherein the memory for storing the validation flag(s) is arranged as a bitmap.

41. A receiver/decoder as claimed in ~~any of claims 29 to 40 for use in the method of claim 17~~ claim 29, further including a protected area of memory for storing a validation code, and wherein the processing means is programmed to decrypt the validation code in such a received MPEG table, to look-up the stored validation code, and to compare the looked-up validation code and the decrypted validation code.

42. An MPEG receiver/decoder for use in performing part of the method of claim 17, comprising:

means for receiving such MPEG tables;

means for storing a public key and an identification for the public key;

a protected area of memory for storing a validation code; and

processing means which is programmed to decrypt the encrypted validation code and signature in such a received MPEG tables using the stored public key corresponding to the private key; to look-up the stored validation code in the protected area of memory; to compare the looked-up validation code and the decrypted validation code; to

generate a signature for the data in the received MPEG table or said part thereof; and to compare the decrypted signature with the signature generated by the receiver/decoder for the received data.

43. A receiver/decoder as claimed in claim 41 ~~or 42~~, wherein the processing means is programmed to inhibit or abort downloading of the data if the looked-up validation code and the decrypted validation code do not match each other.

44. A receiver/decoder as claimed in ~~any of claims 41 to 43~~ claim 41, wherein the memory for storing the validation code is provided by rewritable non-volatile memory.

45. A receiver/decoder as claimed in ~~any of claims 41 to 44~~ claim 41, wherein the memory for storing the validation codes is arranged as a bitmap.

46. A receiver/decoder as claimed in ~~any of claims 29 to 45 for use in a method as claimed in claim 21~~ claim 29, further including a protected area of memory for storing at least one offset, and wherein the processing means is programmed to decrypt the encrypted data block in such a received MPEG table, to look-up said one stored offset in the protected area of memory, and to extract the signature from the decrypted data block using the looked-up offset from the start of the decrypted data block.

47. An MPEG receiver/decoder for use in performing part of the method of claim 22, comprising:

means for receiving such MPEG tables;

means for storing a public key and an identification for the public key;

a protected area of memory for storing at least one offset; and

processing means which is programmed to decrypt the encrypted data block in such a received MPEG table using the stored public key corresponding to the private key; to look-up said one stored offset in the protected area of memory; to extract the signature from the decrypted data block using the looked-up offset from the start of the decrypted data block; to generate a signature for the data in the received MPEG table; and to compare the signature extracted from the decrypted data block with the signature

generated at the receiver/decoder for the received data.

48. A receiver/decoder as claimed in claim 46 ~~or 47~~, wherein at least two such offsets are stored in the protected area of the memory, and the processing means is operable, if the extracted signature and the generated signature do not match, to repeat the looking-up, extracting and comparing using another of the stored offsets.

49. A receiver/decoder as claimed in ~~any of claims 46 or 48~~ claim 46, wherein the memory for storing the offset is provided by rewritable non-volatile memory.

50. A receiver/decoder as claimed in ~~any of claims 29 to 49 for use in a method as claimed in claim 25~~ claim 29, wherein the processing means is programmed to generate a respective module signature for each of the modules in the received module MPEG tables, and to compare each module signature in the received directory MPEG table with the respective module signature generated by the receiver/decoder.

51. An MPEG receiver/decoder for use in performing part of the method of claim 26, comprising:

means for receiving such directory and module MPEG tables;

means for storing a public key and an identification for the public key; and

processing means which is programmed to decrypt the encrypted directory signature in the received directory MPEG table using the stored public key corresponding to the private key; to generate a directory signature for the directory in the received directory MPEG table; to compare the decrypted directory signature and the directory signature generated by the receiver/decoder; to generate a respective module signature for each of the modules in the received module MPEG tables; and to compare each module signature in the received directory MPEG table with the respective module signature generated by the receiver/decoder.

52. A receiver/decoder as claimed in claim 50 ~~or 51~~, wherein the processing means is programmed to inhibit or abort downloading of such a module of the data if the module signature in the received directory MPEG table and the respective module

signature generated at the receiver/decoder for that module do not match each other.

53. A receiver/decoder as claimed in ~~any of claims 29 to 52~~ claim 29, wherein the processing means is programmed to inhibit or abort downloading of the data if the or each decrypted signature and the generated signature do not match each other.

54. A method of downloading at least part of an application to an MPEG receiver/decoder, substantially as described with reference to the drawings.

55. An MPEG receiver/decoder substantially as described with reference to the drawings.

56. An MPEG transmission system substantially as described with reference to the drawings.